

Rec'd PCT/PTO 20 AUG 2004

10/505224

T/JP03/02197

日本国特許庁

27.02.03

JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出願年月日

Date of Application:

2002年 2月28日

REC'D 25 APR 2003

WIPO

PCT

出願番号

Application Number:

特願2002-053575

[ST.10/C]:

[JP2002-053575]

出願人

Applicant(s):

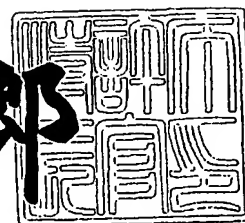
理化学研究所

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

2003年 4月 8日

特許庁長官
Commissioner,
Japan Patent Office

太田信一郎



BEST AVAILABLE COPY

出証番号 出証特2003-3024166

【書類名】 特許願

【整理番号】 P6556

【提出日】 平成14年 2月28日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 15/60

【発明者】

 【住所又は居所】 埼玉県和光市広沢 2 番 1 号 理化学研究所内

 【氏名】 手嶋 吉法

【発明者】

 【住所又は居所】 埼玉県和光市広沢 2 番 1 号 理化学研究所内

 【氏名】 加瀬 究

【発明者】

 【住所又は居所】 埼玉県和光市広沢 2 番 1 号 理化学研究所内

 【氏名】 宇佐見 修吾

【発明者】

 【住所又は居所】 埼玉県和光市広沢 2 番 1 号 理化学研究所内

 【氏名】 牧野内 昭武

【特許出願人】

 【識別番号】 000006792

 【氏名又は名称】 理化学研究所

【代理人】

 【識別番号】 100097515

 【住所又は居所】 東京都港区芝 4 丁目 1 5 番 6 号 ハラビル 2 アサ国際
 特許事務所

 【弁理士】

 【氏名又は名称】 堀田 実

【手数料の表示】

 【予納台帳番号】 027018

 【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9600194

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 境界データのセル内形状データへの変換方法及び変換プログラム

【特許請求の範囲】

【請求項 1】 対象物の境界データからなる外部データ（12）を境界が直交するセル（13）に分割する分割ステップ（A）と、

前記境界データとセルの稜との交点をセル稜切断点とする切断点決定ステップ（B）と、

求めたセル稜切断点を結ぶ境界をセル内形状データとする境界決定ステップ（C）と

分割された各セルを対象物の内側に位置する内部セル（13a）と境界データを含む境界セル（13b）とに区分するセル区分ステップ（D）と、

境界セルを構成するボクセルデータをセル内形状データより内側の内部ボクセルデータと外側の外部ボクセルデータに区分する境界ボクセルデータ区分ステップ（E）とを有する、ことを特徴とする境界データのセル内形状データへの変換方法。

【請求項 2】 前記セルは、境界線が直交する矩形セルであり、

切断点決定ステップ（B）において、境界データとセルの稜との $2^4 = 16$ 通りの交点をセル稜切断点とし、更に回転操作による同値類を同一パターンとして全 6 パターンに区分する、ことを特徴とする請求項 1 に記載の境界データのセル内形状データへの変換方法。

【請求項 3】 前記境界決定ステップ（C）において、前記 6 パターンすべてに対して、セル稜切断点を結ぶ境界線をセル内形状データとする、ことを特徴とする請求項 2 に記載の境界データのセル内形状データへの変換方法。

【請求項 4】 前記境界決定ステップ（C）において、セル内形状データを回転操作と鏡像操作による同値類を同一パターンとして 22 種類のセル内形状パターンに区分する、ことを特徴とする請求項 2 に記載の境界データのセル内形状データへの変換方法。

【請求項 5】 前記セルは、境界面が直交する直方体セルであり、

切断点決定ステップ（Ｂ）において、境界データとセルの稜との $2^{12}=4096$ 通りの交点パターンをセル稜切断点とし、更に回転操作と鏡像操作による同値類を同一パターンとして全144パターンに区分する、ことを特徴とする請求項1に記載の境界データのセル内形状データへの変換方法。

【請求項6】 前記切断点決定ステップ（Ｂ）において、更に反転操作による同値類を同一パターンとしてセル稜切断点が0～6個の全87パターンに区分する、ことを特徴とする請求項5に記載の境界データのセル内形状データへの変換方法。

【請求項7】 前記境界決定ステップ（Ｃ）において、前記全パターンすべてに対して、セル稜切断点を結ぶ境界面をセル内形状データとする、ことを特徴とする請求項5又は6に記載の境界データのセル内形状データへの変換方法。

【請求項8】 対象物の境界データからなる外部データ（12）を境界が直交するセル（13）に分割する分割ステップ（Ａ）と、

前記境界データとセルの稜との交点をセル稜切断点とする切断点決定ステップ（Ｂ）と、

求めたセル稜切断点を結ぶ境界をセル内形状データとする境界決定ステップ（Ｃ）と

分割された各セルを対象物の内側に位置する内部セル（13a）と境界データを含む境界セル（13b）とに区分するセル区分ステップ（Ｄ）と、

境界セルを構成するボクセルデータをセル内形状データより内側の内部ボクセルデータと外側の外部ボクセルデータに区分する境界ボクセルデータ区分ステップ（Ｅ）とを有する、ことを特徴とする境界データのセル内形状データへの変換プログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、形状と物性を統合した実体データを小さい記憶容量で記憶し、CADとシミュレーションを一元化することできる実体データの記憶方法に係り、更に詳しくは、境界データのセル内形状データへの変換方法及び変換プログラムに

関する。

【0002】

【従来の技術】

先端的な研究開発・技術開発の現場では、その高度化・複雑化に伴い、膨大な試行錯誤が不可欠となっており、開発途中でのリスクが高まっている。科学技術立国を目指す我が国として、これらのリスクを極力排し、開発過程の革新的な高度化・効率化を図ることが極めて重要である。

【0003】

現在、研究開発・技術開発の現場において、CAD (Computer Aided Design)、CAM (Computer Aided Manufacturing)、CAE (Computer Aided Engineering)、CAT (Computer Aided Testing) などが、それぞれ設計、加工、解析、試験のシミュレーション手段として用いられている。

また、本発明によって、連続的なシミュレーションであるC-Simulation (Cooperative Simulation)、加工プロセスも考慮したA-CAM (Advanced CAM)、究極の精度が出せるD-fabrication (Deterministic fabrication) など、これから広く普及する見通しである。

【0004】

上述した従来のシミュレーション手段では、対象物を、CSG (Constructive Solid Geometry) やB-rep (Boundary Representation) でデータを記憶している。

【0005】

しかし、CSGでは、対象物全体を微細なソリッドモデルの集合体として記憶するため、データが重くシミュレーション手段（ソフトウェア等）を実装する場合、膨大なデータを扱うこととなり、大型コンピュータを用いた場合でも解析に時間がかかる問題点があった。

【0006】

また、B-r e pでは、対象物を境界で表現するため、データは軽く、データ量は小さくなるが、境界面の内部に関する情報が直接的にはないため、そのままでは変形解析等には適さない問題点があった。

【0007】

更に、これらの従来のデータ記憶手段では、熱・流体解析、固体の大変形解析、これらの連成解析等でその都度、解析に適したメッシュ等に分割して、有限要素法等を適用するため、その解析結果を表示等はできるが、C A Dとシミュレーションを一元化することが困難であり、設計・解析・加工・組立・試験等の各工程を同じデータで管理することができない問題点があった。

【0008】

言い換えれば、現状のS o l i d / S u r f a c e - C A D（以下S-C A Dと呼ぶ）には、以下の問題点があった。

- （１）データが渡らない、内部での変換操作に弱い（数値誤差と処理方法の問題）。
- （２）シミュレーションに直接使えない（内部情報をもっていないのでメッシュを生成しなくてはいけない）。
- （３）C A Mによる加工の検討ができない（最終形状しかもっていない）。

【0009】

また加工においても以下の問題点があった。

- （１）加工プロセスの表現ができない（荒加工や工程設計の支援が不十分）。
- （２）レーザー加工や超先端加工など新しい加工法に対応できていない（切削し
かない、数値精度が足りない）。
- （３）加工法自体の選択ができない（複合体で内部に異なる材料特性を有する）

【0010】

上述した問題点を解決するために、本発明の発明者等は、「形状と物性を統合した実体データの記憶方法」を創案し、出願した（特願2001-25023、未公開）。

【0011】

この発明は、図19に模式的に示すように、対象物の境界データからなる外部データを八分木分割により境界平面が直交する立方体のセルに分割し、分割された各セルを対象物の内側に位置する内部セル13aと境界面を含む境界セル13bとに区分するものである。なおこの図で15は切断点である。

【0012】

この発明により、各セル毎に種々の物性値を記憶し、形状と物性を統合した実体データを小さい記憶容量で記憶することができ、これにより、物体の形状・構造・物性情報・履歴を一元的に管理し、設計から加工、組立、試験、評価など一連の工程に関わるデータを同じデータで管理することができ、CADとシミュレーションを一元化することができる。

【0013】

上述した実体データの記憶方法を以下「ボリウムCAD」又は「V-CAD」と呼ぶ。なお本出願では、V-CADを次のように定義する：「V-CADとは、`voxel dataset`のセル内に境界面が張られたものである」。

【0014】

従来のCADはソリッドと言えども事実上はB-rep等のように中身の無い張りぼてであった。これに対してV-CADは、内部まで詰まっており、物理量も保持できる。内部情報を持つ事により、これまでのB-rep等の形状処理では破綻しやすかった幾何演算が頑健に処理できると期待されている。さらに、V-CADは単なる形状表現の道具としての枠を超え、シミュレーションや加工にも直接活かせるデータ基盤となることを目指している。このような「ものづくり」の体系が真に実現される為には、V-CADを効果的に活かす様なシミュレーション技術や加工技術の開発も併行してなされなければならない。特に加工に関してはこれまで表面形状のデータだけがもっぱら使われていたので、ボリウムデータを真に活かす加工技術は光造形法やラピッドプロトタイピング（3Dインクジェット）を除いて殆ど存在してないとも言える。

【0015】

【発明が解決しようとする課題】

ものづくりの世界の現状を踏まえたとき、従来型のCADが表す形状を読み込

んでV-CAD内にボリュームデータを生成することは非常に重要なことである。このため、V-CADでは、境界セルに形状データ（外部データ）の境界を再構成できる境界データを保有させる必要がある。

【0016】

従来、ボリュームデータからポリゴンを生成する際に一般に用いられているのが、Marching Cubes（以下、MCと略す）である。MCは、例えば、次の文献1に紹介されている。

（文献1）” MARCHING CUBES: A HIGH RESOLUTION 3D SURFACE CONSTRUCTION ALGORITHM”, Computer Graphics, Volume 21, Number 4, July 1987.

参考として、図16に2次元MCのすべての切断点パターンおよび境界線分、図17に3次元MCのすべての切断点パターン（境界面は省略）を示す。

【0017】

3次元MCでは、3次元セル（立方体）の8頂点に正負の数値が書き込まれていて、それをもとに等値面を生成する（以下では値がゼロの面を考える）。立方体の稜の両端の数値の符合が異なるならばその稜上に一つの切断点を設け、符合が同じならば切断点は設けない。それを立方体の12稜について行なったあと、その切断点をもとに面を張るものである。なお2次元MCの場合も同様である。

【0018】

図2は、MCとKTCの矩形セルにおける切断点の相違を示す図の一例である。図2の例において、MCでは、図2aの様な状況では正方形セルの4頂点（白丸）は、いずれも形状（閉じた曲線）の外部に位置するので同じ符合をもち、それゆえセルの4稜（4辺）には切断点が発生することは無く、結果としてこの場合セル内面は全く張られない。これは、MCの立場に立てば、この形状を表すには現在の解像度では粗すぎたということを意味する。従って、この例でMCでは、セルの4稜の切断点を図2b又は図2cのように表現できない問題点があった。

これに対して、後述する本発明（KTC）では、図2bのように4稜には切断

点が2つつ発生し、切断点個数を1稜上で0個または1個に制限すると、図2cのように表現できる。

【0019】

図3は、MCとKTCの稜の切断点の相違を示す図である。MC構築の際にも、切断点個数は1稜上で0個または1個に制限されている。図3aに例示するように、MCではセルの各稜の両端の符合が正負逆の場合に対してのみその稜上に切断点を1つ発生させる。そのため、図3bに示すように、1つの稜上に切断点が与えられたとき、その両端の符合は正負逆の場合のみならず、符合が同じ場合のこともあり、MCではその一部の場合しか、切断点を表現できないことになる。

【0020】

図2において、閉じた実線で示す形状を表現するためには、MCではセルを更に細分する必要がある。その結果、MCを用いたV-CADでは、境界データを保有するためにセルを更に細分する必要がある。その分、形状と物性を統合した実体データを記憶する記憶容量が指数関数的に増大してしまふ。また、記憶容量が増大を避けると、境界部分の形状を精密に表現することが困難となる。

【0021】

本発明は、上述した問題点を解決するために創案されたものである。すなわち、本発明の目的は、Marching Cubes (MC) によるすべてのセル稜切断点を内含し、かつMCでは得られないすべてのセル稜切断点を漏れなく網羅することができ、これによりセル稜切断点を結ぶ境界からなるセル内形状データを漏れなく網羅することができる境界データのセル内形状データへの変換方法及び変換プログラムを提供することにある。

【0022】

【課題を解決するための手段】

本発明によれば、対象物の境界データからなる外部データ(12)を境界が直交するセル(13)に分割する分割ステップ(A)と、前記境界データとセルの稜との交点をセル稜切断点とする切断点決定ステップ(B)と、求めたセル稜切断点を結ぶ境界をセル内形状データとする境界決定ステップ(C)と分割された

各セルを対象物の内側に位置する内部セル（13a）と境界データを含む境界セル（13b）とに区分するセル区分ステップ（D）と、境界セルを構成するボクセルデータをセル内形状データより内側の内部ボクセルデータと外側の外部ボクセルデータに区分する境界ボクセルデータ区分ステップ（E）とを有する、ことを特徴とする境界データのセル内形状データへの変換方法及び変換プログラムが提供される。

【0023】

この方法及び変換プログラムによれば、分割ステップ（A）とセル区分ステップ（D）により、対象物の外部データ（12）を境界が直交するセル（13）に分割したセルの階層として小さい記憶容量で外部データ（12）を記憶することができる。

また、切断点決定ステップ（B）において、境界データとセルの稜との交点をセル稜切断点とするので、「両端の数値の符合が異なるならばその稜上に一つの切断点を設け、符合が同じならば切断点は設けない」MCによるすべての切断点を内含し、かつ境界データとセル稜線とのすべての切断点を漏れなく網羅することができる。

更に、境界決定ステップ（C）において、求めたセル稜切断点を結ぶ境界をセル内形状データとするので、MCによるすべてのセル内形状データを内含し、かつセル内形状データを漏れなく網羅することができる。

また、境界ボクセルデータ区分ステップ（E）において、境界セルを構成するボクセルデータをセル内形状データより内側の内部ボクセルデータと外側の外部ボクセルデータに区分するので、すべてのボクセルデータを隣接するセルとの連続性を保ちながら、内部ボクセルデータと外部ボクセルデータに区分できる。

【0024】

本発明の好ましい実施形態によれば、前記セルは、境界線が直交する矩形セルであり、切断点決定ステップ（B）において、境界データとセルの稜との $2^4=16$ 通りの交点をセル稜切断点とし、更に回転操作による同値類を同一パターンとして全6パターンに区分する。

【0025】

この方法により、2次元セルの場合に、セル稜切断点のパターンをMCによる4パターンを内含する全6パターンに区分し、セル稜切断点のパターンを漏れなく網羅することができる。

【0026】

前記境界決定ステップ(C)において、前記6パターンすべてに対して、セル稜切断点を結ぶ境界線をセル内形状データとする。

また、前記境界決定ステップ(C)において、セル内形状データを回転操作と鏡像操作による同値類を同一パターンとして22種類のセル内形状パターンに区分する。

【0027】

これらの方法により、MCによるすべてのセル内形状データを内含し、かつセル内形状データを漏れなく網羅することができる。

【0028】

本発明の別の好ましい実施形態によれば、前記セルは、境界面が直交する立方体セルであり、切断点決定ステップ(B)において、境界データとセルの稜との $2^{12}=4096$ 通りの交点をセル稜切断点とし、更に回転操作と鏡像操作による同値類を同一パターンとして全144パターンに区分する。

【0029】

この方法により、3次元セルの場合に、セル稜切断点のパターンをMCによる14パターンを内含する全144パターンに区分し、セル稜切断点のパターンを漏れなく網羅することができる。

【0030】

前記切断点決定ステップ(B)において、更に反転操作による同値類を同一パターンとしてセル稜切断点が0～6個の全87パターンに区分する。

この方法により、セル稜切断点が0～6個の全87パターンで、セル稜切断点のパターンを漏れなく網羅することができる。

【0031】

前記境界決定ステップ(C)において、前記全パターンすべてに対して、セル稜切断点を結ぶ境界面をセル内形状データとする。

この方法により、MCによるすべてのセル内形状データを内含し、かつセル内形状データを漏れなく網羅することができる。

【0032】

【発明の実施の形態】

以下、本発明の好ましい実施形態を図面を参照して説明する。

図1は、本発明のデータ変換方法及び変換プログラムのフロー図である。この図に示すように、本発明の方法及び変換プログラムは、分割ステップ(A)、切断点決定ステップ(B)、境界決定ステップ(C)、セル区分ステップ(D)及び境界ボクセルデータ区分ステップ(E)からなる。

【0033】

外部から入力する外部データ12は、多面体を表すポリゴンデータ、有限要素法に用いる四面体又は六面体要素、3次元CAD又はCGツールに用いる曲面データ、或いはその他の立体の表面を部分的な平面や曲面で構成された情報で表現するデータである。

【0034】

外部データ12は、このようなデータ(S-CADデータと呼ぶ)のほかに、(1)V-CAD独自のインターフェース(V-interface)により人間の入力により直接作成されたデータと、(2)測定機やセンサ、デジタイザなどの表面のデジタイズデータや、(3)CTスキャンやMRI、および一般的にVolumeレンダリングに用いられているボクセルデータなどの内部情報ももつVolumeデータであってもよい。

【0035】

分割ステップ(A)では、外部データ取得ステップ(図示せず)で取得した対象物の境界データからなる外部データ12を境界が直交するセル13に分割する。

【0036】

2次元セルの場合に、この分割は四分木分割により境界線が直交する矩形セルに分割する。

【0037】

また3次元セルの場合には、八分木分割により境界平面が直交する立方体のセル13に分割する。この八分木分割では、修正された八分木（オクトリー、Octree）による空間分割を行う。オクトリー表現、すなわち八分木による空間分割とは、目的の立体（対象物）を含む、基準となる立方体13を8分割し、それぞれの領域の中に立体が完全に含まれるか、含まれなくなるまで再帰的に8分割処理を繰り返す。この八分木分割によりボクセル表現よりも大幅にデータ量を減らすことができる。

【0038】

四分木又は八分木による空間分割により分割された一つの空間領域をセル13とよぶ。セルは境界が直交する矩形又は直方体である。矩形又は直方体は、正方形及び立方体が好ましいがこれに限定されない。このセルによる階層構造、分割数もしくは分解能によって空間中に占める領域を表現する。これにより空間全体の中で対象は大きさの異なるセルを積み重ねたものとして表現される。

【0039】

切断点決定ステップ（B）では、境界データとセルの稜との交点をセル稜切断点15とする。

【0040】

2次元セルの場合に、切断点決定ステップ（B）において、境界データとセルの稜との $2^4 = 16$ 通りの交点パターンをセル稜切断点とし、更に回転操作による同値類を同一パターンとして後述する全6パターンに区分する。

【0041】

また3次元セルの場合には、切断点決定ステップ（B）において、境界データとセルの稜との $2^{12} = 4096$ 通りの交点パターンをセル稜切断点とし、更に回転操作と鏡像操作による同値類を同一パターンとして後述する全144パターンに区分する。3次元セルの場合にはパターン数が多いので、更に切断点の有無に関する反転操作による同値類を同一パターンとしてセル稜切断点が0～6個の全87パターンから、切断点7～12個までの全144パターンを導出できる。

【0042】

境界決定ステップ（C）では、求めたセル稜切断点を結ぶ境界をセル内形状デ

ータとする。

【 0 0 4 3 】

2次元セルの場合に、境界決定ステップ（C）において、セル稜切断点の6パターンすべてに対して、セル稜切断点を結ぶ境界線をセル内形状データとする。更に具体的には、セル内形状データを回転操作と鏡像操作による同値類を同一パターンとして後述する22種類のセル内形状パターンに区分するのがよい。

【 0 0 4 4 】

また3次元セルの場合には、境界決定ステップ（C）において、セル稜切断点の全パターンすべてに対して、セル稜切断点を結ぶ境界面をセル内形状データとする。なお、後述する具体例では、セル稜切断点を結ぶ境界面を求めることを「面を張る」と表現する。

【 0 0 4 5 】

セル区分ステップ（D）では、分割された各セルを対象物の内側に位置する内部セル13aと境界データを含む境界セル13bとに区分する。

すなわち本発明では境界セル13bを表現するために修正された四分木又は八分木を使い、完全に内部に含まれるものはその最大の大きさをもつ内部セル13aとし、外部データ12からの境界情報を含むセルは境界セル13bとする。

【 0 0 4 6 】

境界ボクセルデータ区分ステップ（E）では、境界セルを構成するボクセルデータをセル内形状データより内側の内部ボクセルデータと外側の外部ボクセルデータに区分する。

【 0 0 4 7 】

本発明の方法では、必要に応じてステップ（A）～ステップ（E）を繰り返す行う。また、得られたV-CADデータを用いて、例えば、設計・解析・加工・組立・試験等のシミュレーションを順次行い、これを出カステップ（例えばCAMやポリゴンデータとして）に出カする。

また、本発明の変換プログラムは、上述したステップ（A）～ステップ（E）を実行するためのコンピュータプログラムであり、コンピュータに実装して使用する。

【0048】

以下、本発明を更に詳細に説明する。

1. 本発明では、セル内面を生成する為の新しいアルゴリズムを提案する。そのアルゴリズムはMarching Cubeアルゴリズムを完全に包含し、より一般的なアルゴリズムである。

本発明では、ボクセル集合にポリゴンなどの形状を読み込んでボリュームデータを生成するとき、サンプリング点として立方体(セル)の稜と形状表面との交点を切断点として記録し、これら切断点情報をもとに形状表面を復元する。

また本発明では、切断点からのセル内面生成アルゴリズムをまず2次元において完全に定義する。すなわち、切断点配置の同値類の数え上げ(6パターン)およびそれら全切断点配置に対してセル内面が張られる全ての可能性(2²通り)を網羅する。

3次元に対しては、切断点配置に対する同値類の数え上げ(144パターン)を本発明で行った。また、3次元のセル内面生成について、すべての切断点配置の内、過半数のものに対しては、立方体の6面上で切断点を自己交差なく線分で結合していける。このような閉ループが構成されれば、それを外縁とするようなセル内面を比較的容易に張ることが出来る。数え上げの結果、閉ループを有する配置は144パターンの内87パターンである。閉ループを用いて張られたセル内面の实例を後述する。

【0049】

1. 1 従来型CADの表面形状をvoxel datasetに読み込んで、表面とセルの稜との交点(切断点)を記録し、そこからもとの表面形状を三角形面として近似し、稜上切断点から表面形状を復元することが本発明の主題である。

【0050】

1. 2 Marching Cubes (MC)

上述したMarching Cubes (MCと略す)では、セル(立方体)の8頂点に正負の数値が書き込まれていて、それをもとに等値面を生成する(以下では値がゼロの面を考える)。立方体の稜の両端の数値の符合が異なるならば

その稜上に一つの切断点を設け、符合が同じならば切断点は設けない。それを立方体の12稜について行なったあと、その切断点をもとに面を張るというものである。

MCでは、簡潔なアルゴリズムで頑健に面が張れるので、広く世の中で使われるようになった。MCの中身は次の2つにまとめられる。即ち、(1) 切断点配置のパターン分類、および(2) 各切断点配置に対するセル内面の張り方の定義(つまり切断点をどのようにつなぐか)である。MCの曖昧性の問題とは、ここでいう(2)のセル内面の張り方が必ずしも一意でないということであるが、実装上一意に面が張られるように指定することは難しいことではない。それは、曖昧さが生じる場合に対して、値の大きい頂点同士が繋がるようにセル内面を張ると決めれば良い。或いは、値が小さい方の頂点同士を繋ぐと決めても良い。MCの頑健さとは、このように決めることで穴が無く張れる、すなわちセル内面を構成する3角形の稜が2枚の3角形で共有されていることを指す。

【0051】

1. 3 K i t t a C u b e s (K T C)

本発明において、切断点から表面形状を復元する為の新しい広義の手段を提案するが、それをK i t t a C u b e s (以下、K T Cと略す)と名づける。

上述したMCでは、頑健にセル内面が張られるが、そのことは、必ずしも形状を正しく表現していることを意味しない。同じ解像度(=同じセルの大きさ)で形状表現をする際、K T Cを使えばMCよりも正確に表現出来ることが、本発明で明らかになる。

【0052】

K T Cの正確な定義を行なう前に、K T CがMCよりも正確に形状表現出来ることを示す典型的な実例を二次元にて与えておく(図2)。上述したようにMCにおいては、図2 aの様な状況では正方形セルの4頂点は、いずれも形状の外部に位置するので同じ符合をもち、それゆえセルの4稜には切断点が発生すること無く、結果としてこの場合セル内面は全く張られない。これは、MCの立場に立てば、この形状を表すには現在の解像度では粗すぎたということを意味する。

次に同じ解像度に対してK T Cを適用する。切断点のチェックにより図2 bの

様な切断点が記録される。1稜上の切断点個数を1個までとし図2 bに現れた2個の切断点を適当な位置の1個で代表させれば、最終的に図2 cの様なセル内面が張られることになる。両者の表現力の差は明白である。

【0053】

実は、アルゴリズムの枠組みをKTCとMCとで比較した場合、KTCはMCを完全に包含していることが後述される。つまり、MCはKTCのサブセットである。その意味で、KTCの形状表現がMCよりも豊かであるのは当然である。

図2において、1稜上の切断点を1個までに制限したが、MCとKTCの枠組みを理解する為には、この制限を明確に定義することが重要である。MC構築の際にも、切断点個数は1稜上で0個または1個に制限されていた。本発明においても、切断点の個数は1稜上で0個または1個の枠組みでKTCを構築する。両者の差は、MCではセルの各稜の両端の符合が正負逆の場合に対してのみその稜上に切断点を1つ発生させるが（図3 a）、KTCでは1つの稜上に切断点を与えられたとき、その両端の符合は正負逆の場合のみならず、符合が同じ場合のこともある（図3 b）。

【0054】

以下、MCの場合を「狭義1稜1切断点条件」、KTCの場合を「広義1稜1切断点条件」と呼び、これらを明確に区別することにする。図3 aと図3 bから明らかな様に、広義1稜1切断点条件は狭義1稜1切断点条件を完全に含んでいる。したがって、両者から与えられる全ての切断点配置のパターンに関しても、MCの全パターンは、KTCの全パターンの中に完全に含まれる。

【0055】

2. 二次元K i t t a C e l l s (2D-KTC) KTCを2次元において構成する。KTCは2次元においては、K i t t a C e l l sの略称とする。

【0056】

2. 1 2D-KTCにおける切断点配置の分類

2D-KTCは、2段階で構成される。第1段階は、正方形の4稜に対する切断点配置の分類であり、第2段階は各切断点配置に対するセル内線（2次元なので、面ではなく線）の張り方の定義である。

図4は、2D-KTCにおける切断点配置のすべてのパターンを示す図である。まず、第1段階において、図3bの広義1稜1切断点条件のもとで、正方形の4本の稜に対する切断点配置の分類を行う。4稜の各々に対する、切断点の存在または非存在の2つの可能性から $2^4 = 16$ 通りであるが、それらを回転操作に対する同値類を数え上げると表1および図4の6パターンにまとめられる。なおこの図で丸で囲んだものは、2D-MCに存在するものを示している。

切断点の存在非存在の反転操作を同値類の分類に加えるならば、6パターンは4パターンに減る。ただし、2.2で面の張り方を定義する際には、6パターン全てに対して定義する必要がある。

【0057】

【表1】

Number of Cut Edges: k	0, 4	1, 3	2	Total ($k=0-4$)
All Configurations of Cut Edges: $\binom{4}{k}$	1	4	6	16
Equivalence Class	1	1	2	6(4)
Index in Fig. 4	0a(4a)	1a(3a)	2a, 2b	—

Table 1: Equivalence class of configurations of cut edges in 2D-KTC

【0058】

後述するAppendix A, Bにおいて2Dおよび3DのMCを扱うが、MCにおいて頂点の白黒塗りわけ（これは頂点の値の正負の区別に対応）の白黒反転が分類上許されるのは、白黒反転後の切断点配置は、白黒反転前の切断点配置と不変だからである。

図4の各配置の縮退度は、順に1, 4, 4, 2, 4, 1である（足し合わせると16）。ここで、「縮退度」とは、回転操作に対する同値類の数である。

【0059】

2.2 2D-KTCにおける面張り定義

2D-KTCの第2段階に進む。ここでは、2.1で得られた6種類の切断点配置の各々に対して、線の張り方のあらゆる可能性を網羅する。網羅のポイントは、「注目している正方形内にセル内線を張る際に、その正方形の稜上に存在する全ての切断点を利用する必要はない」ということである。一つの稜は、2つの

正方形に共有されており、その稜に切断点があったとき、もし片方の正方形内でそれがセル内線を張るために利用されなかったとしても、他方のセルで利用される可能性がある。

もちろん、どちらの正方形からも利用されない可能性もある。それは、非常に小さい形状が、その1本の稜とだけ交差する場合である。それに対する実装上の処理としては、切断点情報を活かす態度としては、その稜上に微小な線分を置くことになるだろうし、別の態度は、その切断点を線張りに全く利用しないことである。ただし、線分を置く事や3次元にて面分を置くことは、形状表面が多様体でなくなることを意味するので、本発明中では網羅の対象としない。同じ理由で、セル内面同士が交差する場合（具体的には、正方形が直交する2枚のセル内線により1辺の長さが元の正方形の $1/2$ の正方形へと4等分される場合）も数え上げの対象から除外した。

【0060】

図5は、2D-KTCにおける線張りのすべてのパターンを示す図であり、2.1で得られた6種類の切断点配置の各々に対して、線張りのあらゆる可能性を調べた結果である。なおこの図で丸で囲んだものは、2D-MCに存在するものを示している。

これらの同値類 (equivalence class) を数え上げる際には回転操作のみならず鏡像操作も利用された。セル内線が途中で途切れたりしないことを仮定すれば、図5の4頂点を白と黒で塗り分けることが出来る。MCの場合と違って、この塗りわけはセル内線を張った後に行われる。図5のセル内線だけに注目すると、例えば2A(1), 3(5), 4(9)のように同じタイプのものが現れるが、全ての切断点配置に対する面の張り方を網羅する目的の為に、どちらも省略できない。ちなみに図5に描かれた22種類のパターンをセル内線の形状で分類すれば10ケースになる。

【0061】

3. 三次元K i t t a C u b e s (3D-KTC)

3. 1 3D-KTCにおける切断点配置の分類

総数 $2^{12} = 4096$ の切断稜配置が、回転操作と鏡像操作のみを許したとき、

いくつかの同値類にまとまるか、またそれらはどのような配置であるかを解き明かす。

まず、[数1]の(1)(2)式が成り立つから、切断点個数が0～5個に対する配置に切断点の有無に関する反転操作をすることで切断点個数が7～12個に対する配置が得られることがわかり、同値類に関してもその反転操作で対応づいている。

【0062】

【数1】

$$2^{12} = \sum_{k=0}^{12} \binom{12}{k} \quad \dots(1)$$

$$\binom{12}{k} = \binom{12}{12-k} \quad \dots(2)$$

【0063】

まず図6のように立方体の辺に1～12までのインデックスを与えておく。正八面体群の回転操作24に鏡像操作を加えると48の群の操作が可能である。4096個の配置の指標を1番目から順に一つずつ新出か既出かをチェックして行くが、そのとき個々の配置に48回の操作を施し、既出のリストと照らすことでそれと同じ配置が既に登場しているかどうか判断できる。表2に示すように、最終的に144種類の互いに異なる切断点配置が得られる。

【0064】

【表2】

Number of Cut Edges: k	0, 12	1, 11	2, 10	3, 9	4, 8	5, 7	6	Total ($k=0-12$)
All Configurations of Cut Edges: $\binom{12}{k}$	1	12	66	220	495	792	924	4096
Equivalence Class	1	1	4	6	18	24	30	144(87)

Table 2: Equivalence class of configurations of cut edges in 3D-KTC

【0065】

セル内面の張り方の定義は、これら全配置に対して行われるべきものであるが、前段落で述べたように切断点個数が7～12個に対する同値類は、切断点個数が0～5個に対する同値類から一意に定まるので、切断点個数が0～6個に対する全87パターンを表3にまとめ、図7～9には図を掲げる。なおこれらの図で

丸で囲んだものは、3D-MCに存在するものを示している。

【0066】

【表3】

♣	0 Cut Edge	4-13	(1 2 9 12)	♣	6 Cut Edges
0-1	(nil)	4-14	(1 3 8 8)	6-1	(1 2 3 4 5 6)
♣	1 Cut Edge	4-15	(1 3 8 9)	6-2	(1 2 3 4 5 7)
1-1	(1)	4-16	(1 3 8 10)	6-3	(1 2 3 4 5 9)
♣	2 Cut Edges	4-17	(1 3 6 11)	6-4	(1 2 3 4 5 11)
2-1	(1 2)	4-18	(1 3 6 12)	6-5	(1 2 3 4 9 10)
2-2	(1 3)	♣	5 Cut Edges	6-6	(1 2 3 4 9 12)
2-3	(1 6)	5-1	(1 2 3 4 5 10)	6-7	(1 2 3 5 6 7)
2-4	(1 7)	5-2	(1 2 3 4 9)	6-8	(1 2 3 5 6 8)
♣	3 Cut Edges	5-3	(1 2 3 5 6)	6-9	(1 2 3 5 6 9)
3-1	(1 2 3)	5-4	(1 2 3 5 7)	6-10	(1 2 3 5 6 10)
3-2	(1 2 5)	5-5	(1 2 3 5 8)	6-11	(1 2 3 5 7 8)
3-3	(1 2 7)	5-6	(1 2 3 5 9)	6-12	(1 2 3 5 7 9)
3-4	(1 2 9)	5-7	(1 2 3 5 10)	6-13	(1 2 3 5 7 10)
3-5	(1 3 6)	5-8	(1 2 3 5 11)	6-14	(1 2 3 5 7 11)
3-6	(1 3 9)	5-9	(1 2 3 5 12)	6-15	(1 2 3 5 7 12)
3-7	(1 3 10)	5-10	(1 2 3 7 8)	6-16	(1 2 3 5 8 9)
3-8	(1 3 12)	5-11	(1 2 3 7 9)	6-17	(1 2 3 5 8 10)
3-9	(1 7 12)	5-12	(1 2 3 7 10)	6-18	(1 2 3 5 8 11)
♣	4 Cut Edges	5-13	(1 2 3 8 9)	6-19	(1 2 3 5 8 12)
4-1	(1 2 3 4)	5-14	(1 2 3 8 10)	6-20	(1 2 3 5 11 12)
4-2	(1 2 3 5)	5-15	(1 2 3 9 11)	6-21	(1 2 3 7 8 10)
4-3	(1 2 3 7)	5-16	(1 2 3 9 12)	6-22	(1 2 3 7 9 10)
4-4	(1 2 3 8)	5-17	(1 2 3 10 12)	6-23	(1 2 3 7 9 12)
4-5	(1 2 3 9)	5-18	(1 2 5 7 11)	6-24	(1 2 3 7 10 12)
4-6	(1 2 3 10)	5-19	(1 2 5 7 12)	6-25	(1 2 3 8 9 11)
4-7	(1 2 5 6)	5-20	(1 2 7 8 9)	6-26	(1 2 3 8 9 12)
4-8	(1 2 5 7)	5-21	(1 2 7 9 12)	6-27	(1 2 3 8 10 12)
4-9	(1 2 7 8)	5-22	(1 2 7 10 12)	6-28	(1 2 7 8 9 12)
4-10	(1 2 7 9)	5-23	(1 3 6 8 9)	6-29	(1 3 6 8 9 12)
4-11	(1 2 7 10)	5-24	(1 3 6 8 10)	6-30	(1 3 6 8 10 11)
4-12	(1 2 9 11)	—	—	—	—

Table 3: Contents of equivalence class of cut edges configurations. All figures are in Fig.6

【0067】

144種類の配置のどれもが、実際に表面データをボクセル集合に読み込んだ際の切断点配置として出現し得るものである。Appendix Bの表7に3D-MCの全ての切断配置を3D-KTCのそれと対応させた。逆に、表7に現れてないすべてのKTCの切断点配置は、MCには存在しない。

【0068】

3. 2 3D-KTCにおける面張り定義に関して

与えられた全ての切断点を結ぶ線分が立方体の表面上で閉ループを構成する一

例を図 1 0 に示した。図 6 に立方体と共に示されたのは、準正多面体の一つであり正 3 角形 8 枚と正方形 6 枚とからなる 1 4 面体である。切断点の位置が各稜の中心点であるとしても切断点配置の数え上げや閉ループを考察する上で一般性を失わない。この準正多面体は、それら 1 2 稜上の中心点 1 2 個に対する凸包となっている。上記の閉ループを辿る為の道としては、1 4 面体の各正方形面の 2 本ずつの対角線つまり全部で 1 2 本の対角線と、1 4 面体がもともと有する 2 4 稜とを合わせた 3 6 本である。

この準正多面体の頂点 1 2 個と稜 3 6 本のパスとの関係を平面上のグラフとして表現したものが図 1 1 である。この図を元に非自己交差ハミルトン閉路の数え上げを説明する。このグラフにおいて 1 2 個の頂点は全て価数 6 で同等となっており、ここでの数え上げは、特定の一つの頂点から行うだけでよい。許されるパスは、立方体の表面上で与えられた全ての切断点を通して、途中で同じ稜や同じ頂点を 2 度以上通らずに始点に戻ってくるものである。切断点個数が 0 ~ 1 2 に対する切断点配置全 1 4 4 種類のうちの過半数である 8 7 種類がこのような非自己交差ハミルトンパスを持つことを数え上げた。

【0 0 6 9】

実は、これら閉ループから生成されるセル内面の多くは、V-CAD にとって好ましいものとなっている。V-CAD が、セル内の情報として物理情報を持つことは上述したが、多媒質にも対応できる。そのとき異材の界面では、セル内を複数の物質が占めることとなるが、この物質の数が 3 種類以上になったとき、何らかの処理を施すことにより 1 つのセル内の物質数は 2 種類までに制限することが好ましい。なお閉ループから作るセル内面の多くは、セルを 2 分割するので、好都合である。

【0 0 7 0】

ただし、全ての閉ループがセルを 2 領域に分けるのではない。たとえば、図 1 2 の様に閉ループが作れてもセルが分割されないこともある。図 1 3 の例は、閉ループが作れてもそれを縁とする 3 角形分割の仕方によっては 2 分割にも 3 分割にも成り得ることを示している。逆に閉ループを作れなくてもセルを 2 分割する場合もあり、その一つを図 1 4 に示す。

【0071】

3.3 三角形分割に関して

閉ループが出来ると、それを縁とするようなセル内面を張ることは多くの場合容易である。セル内面を三角形分割する問題は、平面上で多角形を三角形分割する問題と関係しており、その三角形分割のパターンの場合の数は、カタラン数を用いて表されることが知られている。カタラン数と三角形分割に関しては、Appendix Cで後述する。

【0072】

3.4 3D-KTCの実装例

閉ループを利用した実装を試みた。B-rep形状表現を読み込んで切断点を記録し、それらに対して閉ループを探し、三角形分割を行った。例題は、サイクリッド（図15a）および金型（図15b）である。これらを切断点情報をもとに復元したのが、図15c, dである。これらは、KTCの面張りの可能性のごく一部分としての、閉ループから生成したものだが、十分な表現力を持っているといえる。

ここで強調しておくべきことは、これら閉ループと対応する切断点配置は、KTCの一部に過ぎないが、それでもMCの切断点配置の枠組みにはおさまり切れないものだという事である。MCの切断点配置が全て合わせても14通りであるのに対して、KTCの切断点配置144種類の内、閉ループと対応づくものだけでも前述のように87種類ある。1例を示すと、図13の切断点配置は閉ループを構成可能であるが、MCの中には存在しないパターンである。

【0073】

4. まとめ

上述のように、KTCとMCを同じ解像度で比較するとき、KTCの方がMCよりも精密な形状表現が可能である。これは、KTCのアルゴリズムの枠組みがMCのそれを完全に含んでいることから当然の結果である。

本発明では広義の1稜1切断点だけを仮定したときにKTCが示す多様さを厳密に提示した。

【0074】

また、ここで示した実装例では、切断点から最初に見つけた閉ループを使い、その内部の三角形分割化は一つのパターンだけをあてはめたという単純な方法であるにも拘わらず、形状を綺麗に復元できている。これはKTC面張りの可能性のごく一部分をV-CADに用いた結果として位置付けられる。

【0075】

Appendix A: 二次元Marching Cubes (2D-MC)

図16に2D-MCの同値類4種を示す。正方形の4頂点上の白黒反転操作と回転操作だけを用いて16 ($=2^4$) 種類が、この4パターンに分類される。鏡像操作は必要では無かった。図16中各パターンにmc-0, mc-1, mc-2, mc-3のインデックスを割り当てた。各パターンが持つ縮退度は、順に2, 8, 4, 2であり、これらを全て足すと16に戻ることが確認される。

2. 1で2D-KTCを述べたが、同値類を分類する前の配置の総数は、2次元においてはMCもKTCも同じ16 ($=2^4$) であることに注意されたい。これは、正方形の頂点数と辺の数がどちらも4であることに起因する。(立方体では、頂点と稜の個数が異なるので、配置の総数も3D-MCの256 ($=2^8$) に対して、3D-KTCは4096 ($=2^{12}$) と異なる。)

表4は切断点配置に関する同値類の個数を、全配置数と対応させた表である。また表5は、MCとKTCの対応表であり。ここに登場しないKTC、つまり図4の1, 3は、MCには存在しない。

【0076】

【表4】

Number of Black Vertices: k	0(or 4)	1(or 3)	2	Total($k=0-4$)
All Configurations: $\binom{4}{k}$	1	4	6	16
Equivalence Class	1	1	2	4

Table 4: Equivalence class of cut edges configurations in 2D-MC

【0077】

【表 5】

2D-MC Index	0	1	2	3
2D-KTC Index	0	2A	2B	4

Table 5: Correspondence between 2D-MC in Fig.13 and 2D-KTC in Fig.3

【0078】

Appendix B: 三次元Marching Cubes (3D-MC)

図17にMCの同値類を全て示した。分類に用いられた操作は、頂点上の白黒反転操作、回転操作、そして鏡像操作である。その結果、256 ($=2^8$) 種類の頂点塗り分けは、これら14パターンに分類される。これらを切断点の個数に関してまとめると表6のようになる。

【0079】

【表 6】

Number of Black Vertices: k	0(or 8)	1(or 7)	2(or 6)	3(or 5)	4	Total($k=0-8$)
All Configurations of Black Points: $\binom{8}{k}$	1	8	28	56	70	256
Equivalence Class	1	1	3	3	6	14

Table 6: Equivalence class of configurations of cut edges configurations in 3D-MC

【0080】

MCの図は、通常MCの原論文に合わせて0-14の計15通りで表されるが、ここには14通りしかないと注意されたい(図17)。省略されたMC-14は、MC-11の鏡像対称であった。MC-11の面の張り方が決まれば鏡像操作によりMC-15のそれも一意に決まるので、本発明内での記述の一貫性(同値類の数え上げに鏡像操作も用いること)を保つ上で省略する。

これら切断点に対する面張りの定義は、MCの原論文においては、1つの切断点配置に対して1つずつ用意されたが、必ずしも一意に定まるわけでない。

図17は原論文の図の並びと同じであるが、セル内面が省かれ、上述の理由でMC-14が削除されている。これらMCの全ケースとKTCとの対応を表7に示す。

【0081】

【表7】

3D-MC Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13
3D-KTC Index	0-1	3-6	4-12	8-6	6-29	5-18	7-18	9-8	4-7	6-30	8-7	6-28	8-12	12-1

Table 7: Correspondence between 3D-MC in Fig.14 and 3D-KTC in Fig.6.

【0082】

Appendix C: 多角形の3角形分割とカタラン数

立方体の表面を辿る閉ループを縁とするような折れ面の生成、すなわちループの内部を3角形分割する問題は、多くの場合平面上の凸 n 角形を3角形分割する問題に帰着できる。（閉ループ全体が平面的では無い場合、すなわち折れ曲がりの激しいセル内面が閉ループから生成される場合には、後者に帰着できない恐れがあるかもしれない。）

後者の場合の数を求める問題は、カタラン数と関連していることが知られている。結果のみを以下に記す。

カタラン数 C_m は（数2）の（3）式で示される。このとき、凸 n 角形をその内部で交わらない対角線によって3角形分割するとき、異なる分割の仕方が幾通りかを求めたい。求める数 T_n は（数2）の（4）式で与えられる。

【0083】

【数2】

$$C_n = \frac{1}{n+1} \binom{2n}{n} \quad \dots(3)$$

$$T_n = C_{n-2} = \frac{1}{n-1} \binom{2(n-2)}{n-2} \quad \dots(4)$$

【0084】

この式に、 $n=3, 4, \dots, 12$ を代入すると、 $T_3=1, T_4=2, T_5=5, T_6=14, \dots, T_{12}=16796$ が得られる。

$n=5$ 以下はさほど難しくない。 $T_6=14$ について補足する。正六角形に対して、これら14通りを同値類にまとめると図18に示した3パターンにまとまる。図18aと図18bは回転操作によって現れる6通りおよび2通りを代表し

、図18cは回転操作と鏡像操作によって現れる6通りを代表している。これらを足し合わせると14通りになることが確認される。

【0085】

【発明の効果】

上述したように、本発明の方法及び変換プログラムによれば、分割ステップ（A）とセル区分ステップ（D）により、対象物の外部データ（12）を境界が直交するセル（13）に分割したセルの階層として小さい記憶容量で外部データ（12）を記憶することができる。

また、切断点決定ステップ（B）において、境界データとセルの稜との交点をセル稜切断点とするので、「両端の数値の符合が異なるならばその稜上に一つの切断点を設け、符合が同じならば切断点は設けない」MCによるすべての切断点を内含し、かつ境界データとセル稜線とのすべての切断点を漏れなく網羅することができる。

【0086】

更に、境界決定ステップ（C）において、求めたセル稜切断点を結ぶ境界をセル内形状データとするので、MCによるすべてのセル内形状データを内含し、かつセル内形状データを漏れなく網羅することができる。

また、境界ボクセルデータ区分ステップ（E）において、境界セルを構成するボクセルデータをセル内形状データより内側の内部ボクセルデータと外側の外部ボクセルデータに区分するので、すべてのボクセルデータを隣接するセルとの連続性を保ちながら、内部ボクセルデータと外部ボクセルデータに区分できる。

【0087】

従って、本発明の境界データのセル内形状データへの変換方法及び変換プログラムは、MCによるすべてのセル稜切断点を内含しかつMCでは得られないすべてのセル稜切断点を漏れなく網羅することができ、これによりセル稜切断点を結ぶ境界からなるセル内形状データを漏れなく網羅することができる、等の優れた効果を有する。

【図面の簡単な説明】

【図1】

本発明のデータ変換方法及び変換プログラムのフロー図である。

【図 2】

MCとKTCの矩形セルの切断点の相違を示す図である。

【図 3】

MCとKTCの稜の切断点の相違を示す図である。

【図 4】

2D-KTCにおける切断点配置のすべてのパターンを示す図である。

【図 5】

2D-KTCにおける線張りのすべてのパターンを示す図である。

【図 6】

準正多面体の一つであり正3角形8枚と正方形6枚とからなる14面体の図である。

【図 7】

3D-KTCにおける切断点配置のうち切断点個数が0～4個の場合の30パターンを示す図である。

【図 8】

3D-KTCにおける切断点配置のうち切断点個数が4又は5個の場合の27パターンを示す図である。

【図 9】

3D-KTCにおける切断点配置のうち切断点個数が6個の場合の全30パターンを示す図である。

【図 10】

与えられた全ての切断点を結ぶ線分が立方体の表面上で閉ループを構成する例を示す図である。

【図 11】

図6の準正多面体の頂点12個と稜36本のパスとの関係を平面上のグラフとして表現したものである。

【図 12】

閉ループが作れてもセルが分割されない例を示す図である。

【図 1 3】

閉ループが作れてもそれを縁とする 3 角形分割の仕方によっては 2 分割にも 3 分割にも成り得る例を示す図である。

【図 1 4】

閉ループを作れなくてもセルを 2 分割する例を示す図である。

【図 1 5】

サイクライドと金型の B-r e p 形状表現と K T C の面張りとを比較したディスプレイ上の中間調画像である。

【図 1 6】

2 D-MC の同値類 4 種を示す図である。

【図 1 7】

3 D-MC の同値類を全てを示す図である。

【図 1 8】

正 6 角形における 1 4 通りのカタラン数を 3 パターンで示す図である。

【図 1 9】

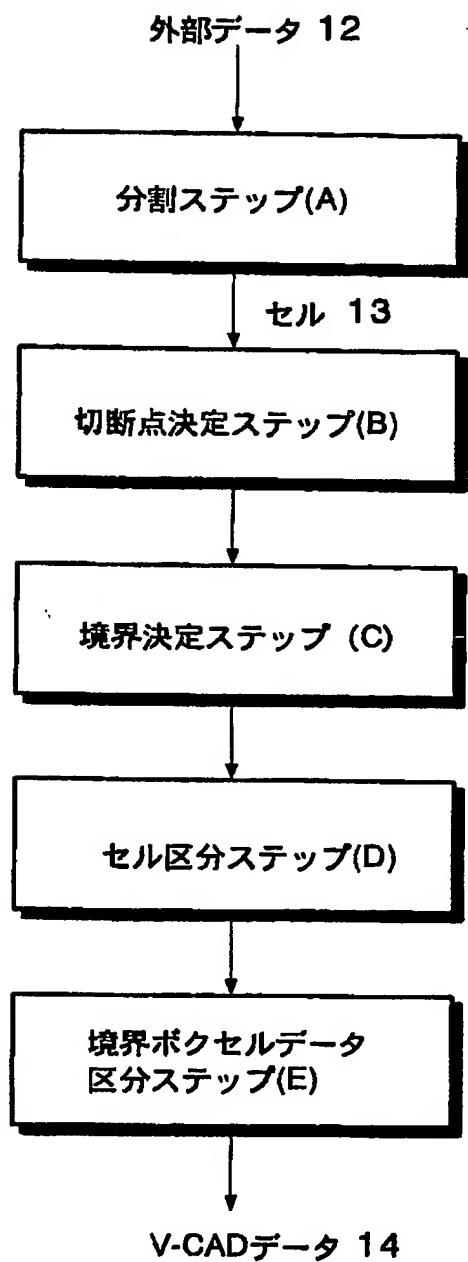
V-CAD の原理図である。

【符号の説明】

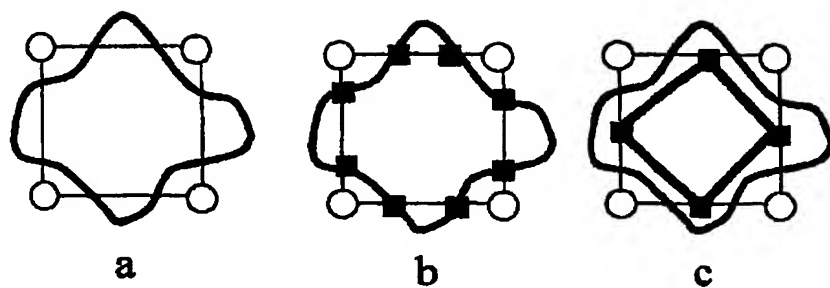
- 1 2 外部データ、1 3 セル、
- 1 3 a 内部セル、1 3 b 境界セル、
- 1 4 V-CAD データ、1 5 切断点

【書類名】 図面

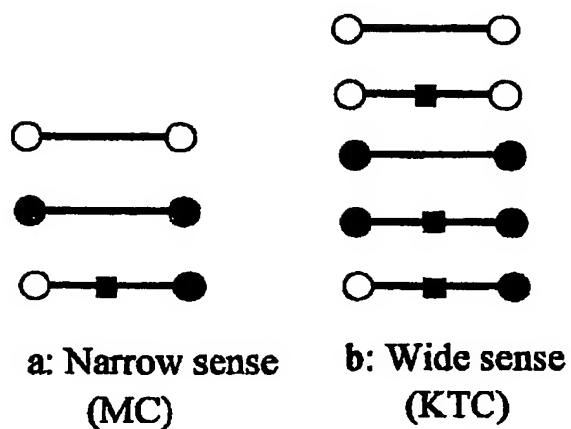
【図 1】



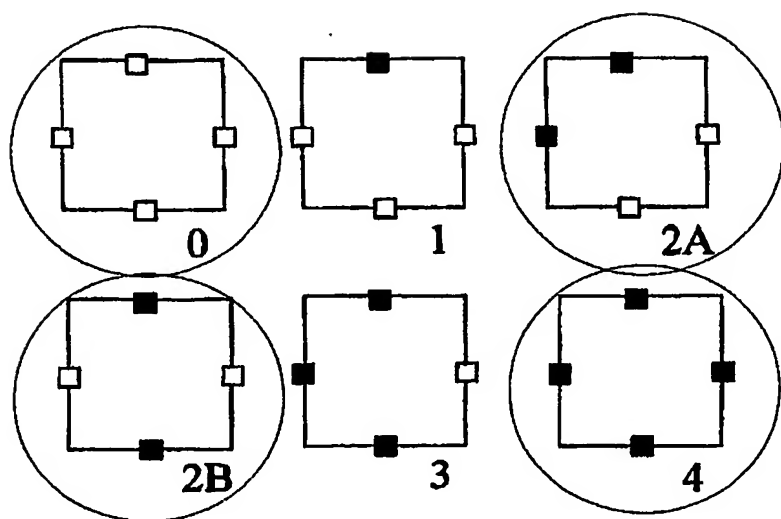
【図 2】



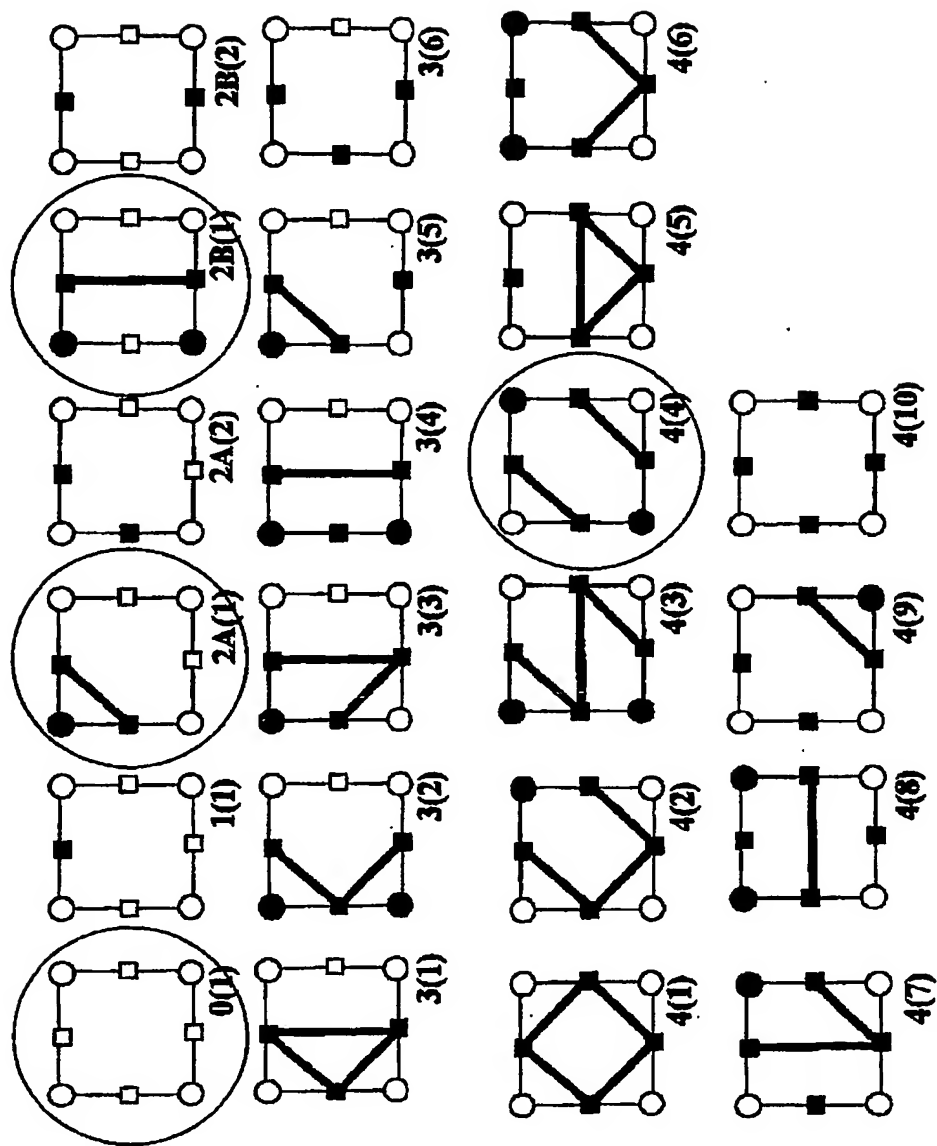
【図 3】



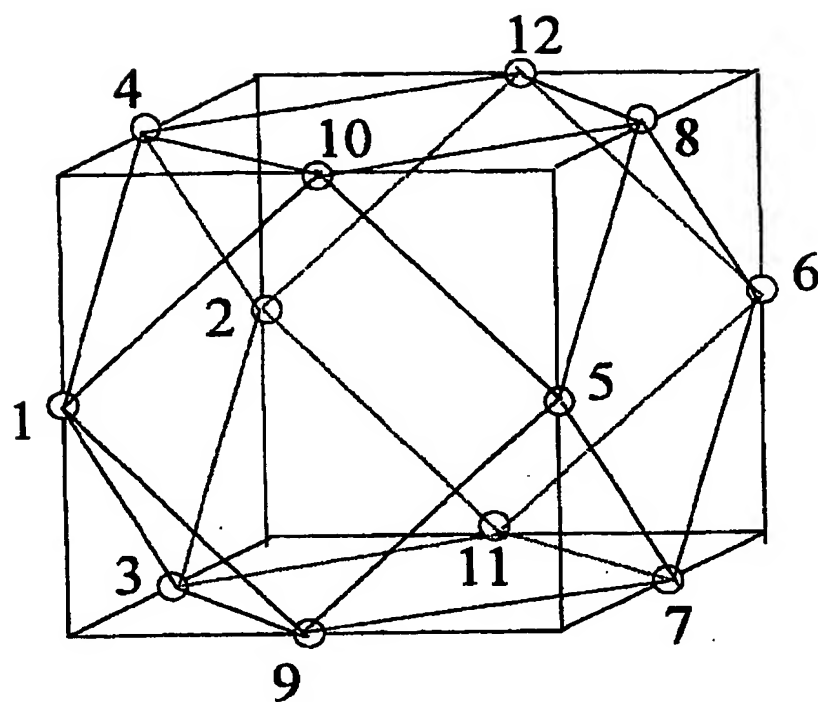
【図 4】



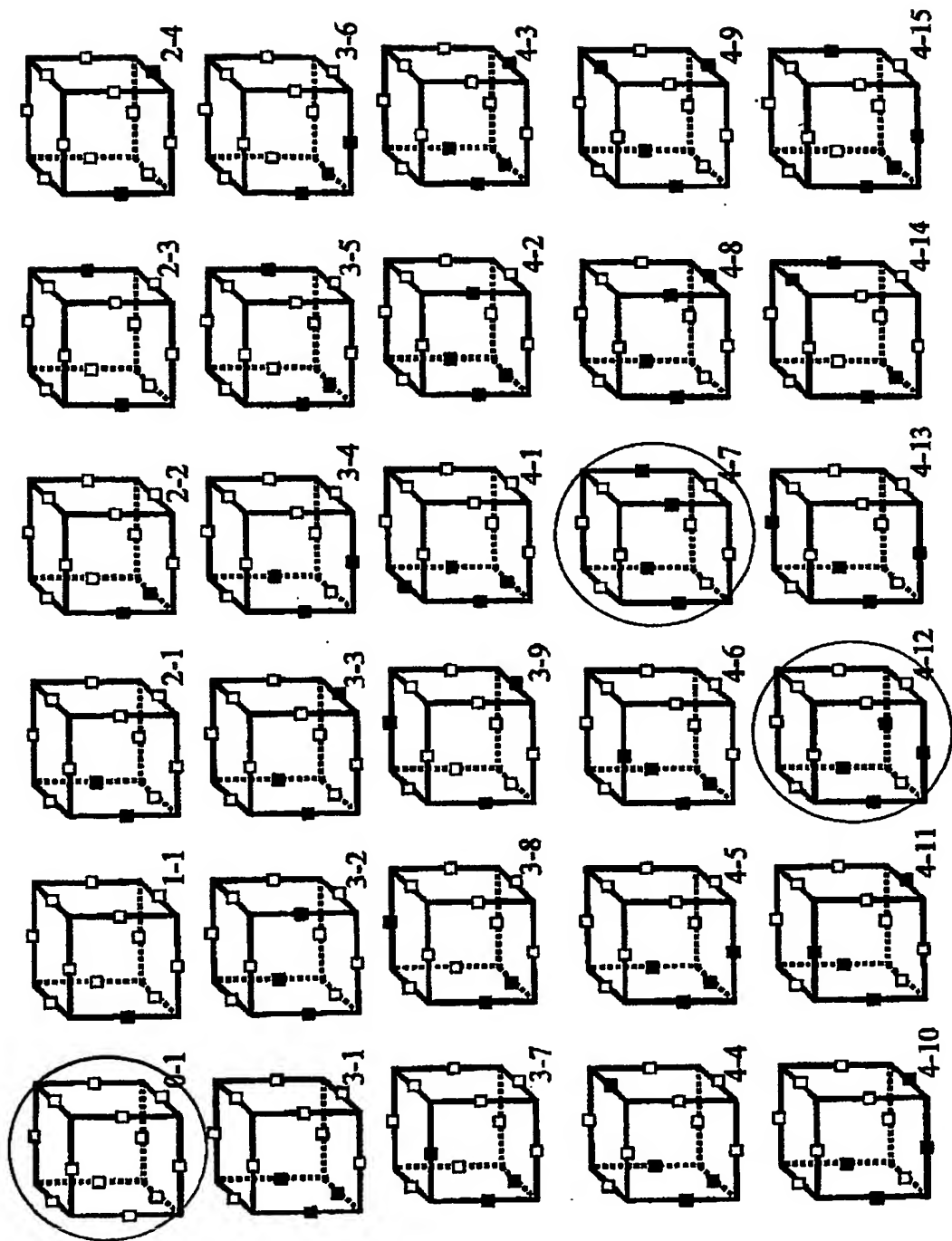
【図 5】



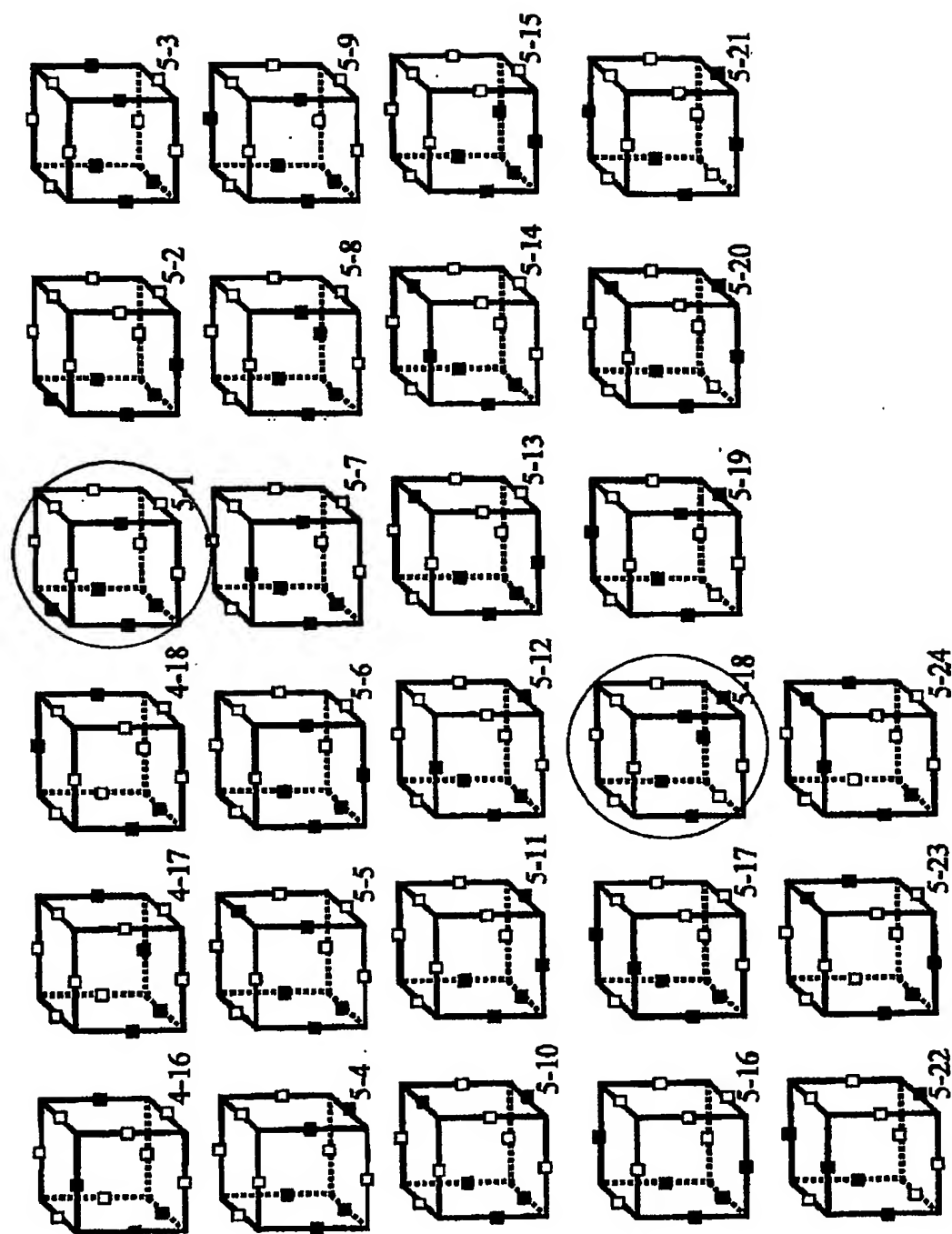
【図6】



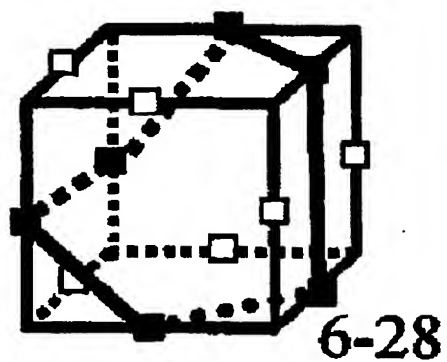
【図7】



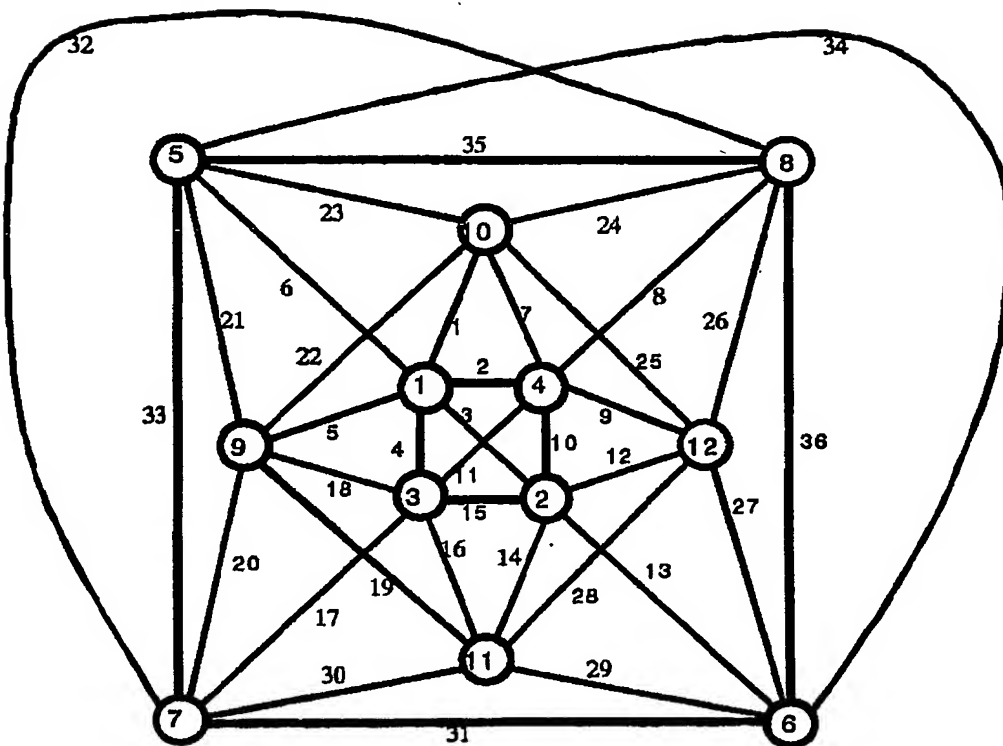
【図8】



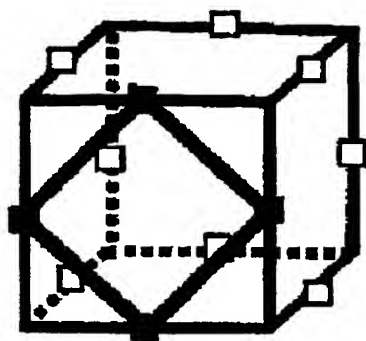
【図10】



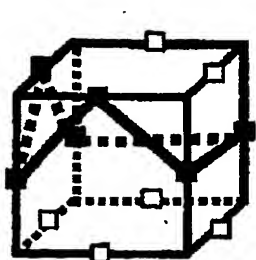
【図11】



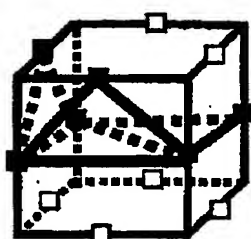
【図 12】



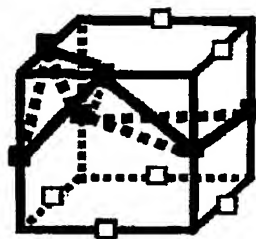
【図 13】



a

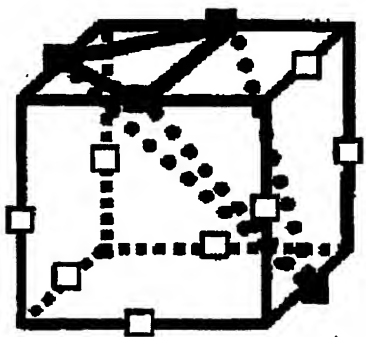


b



c

【図 14】



【図 15】

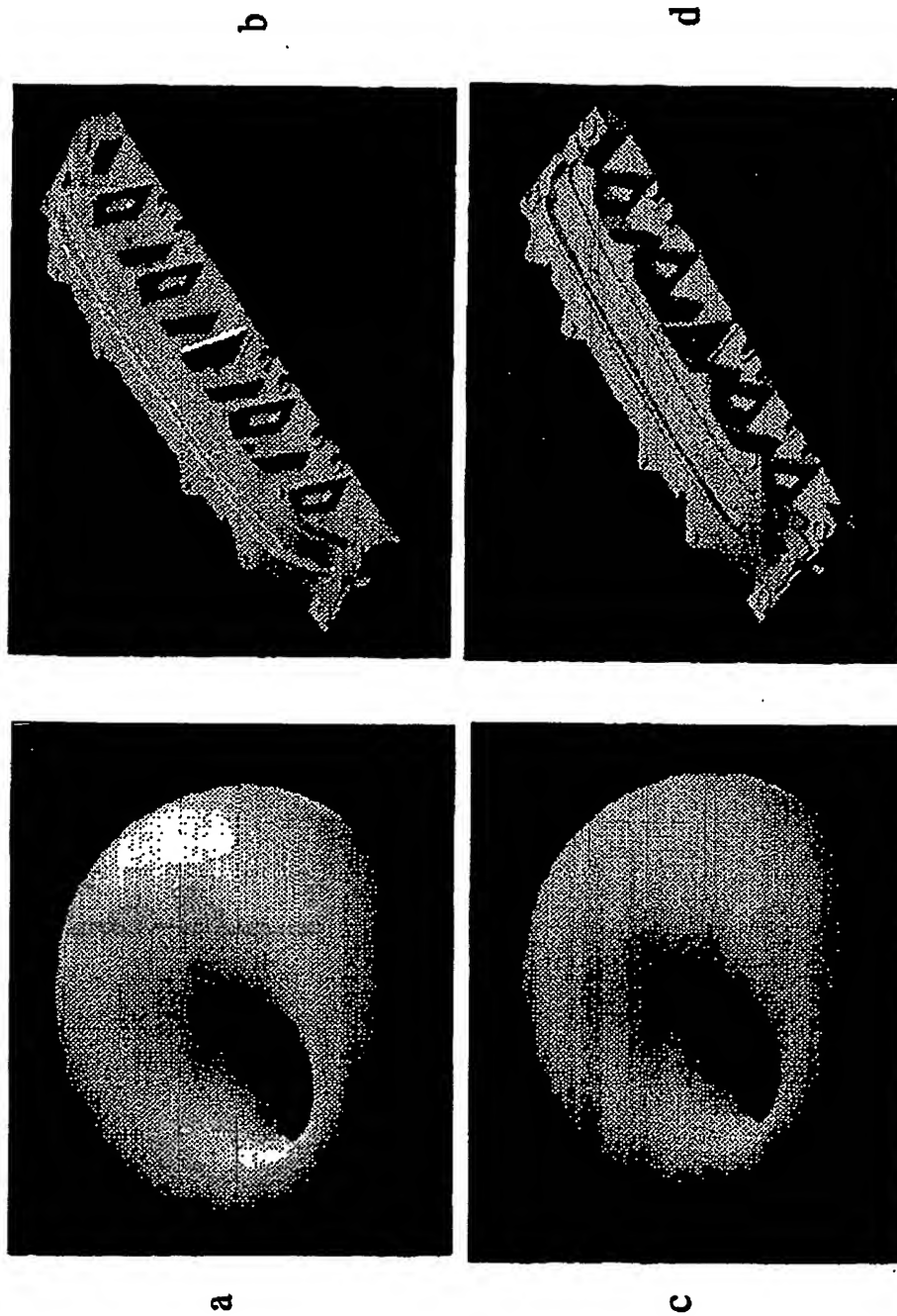
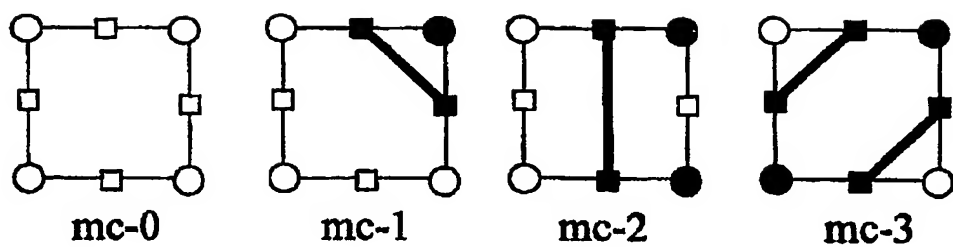
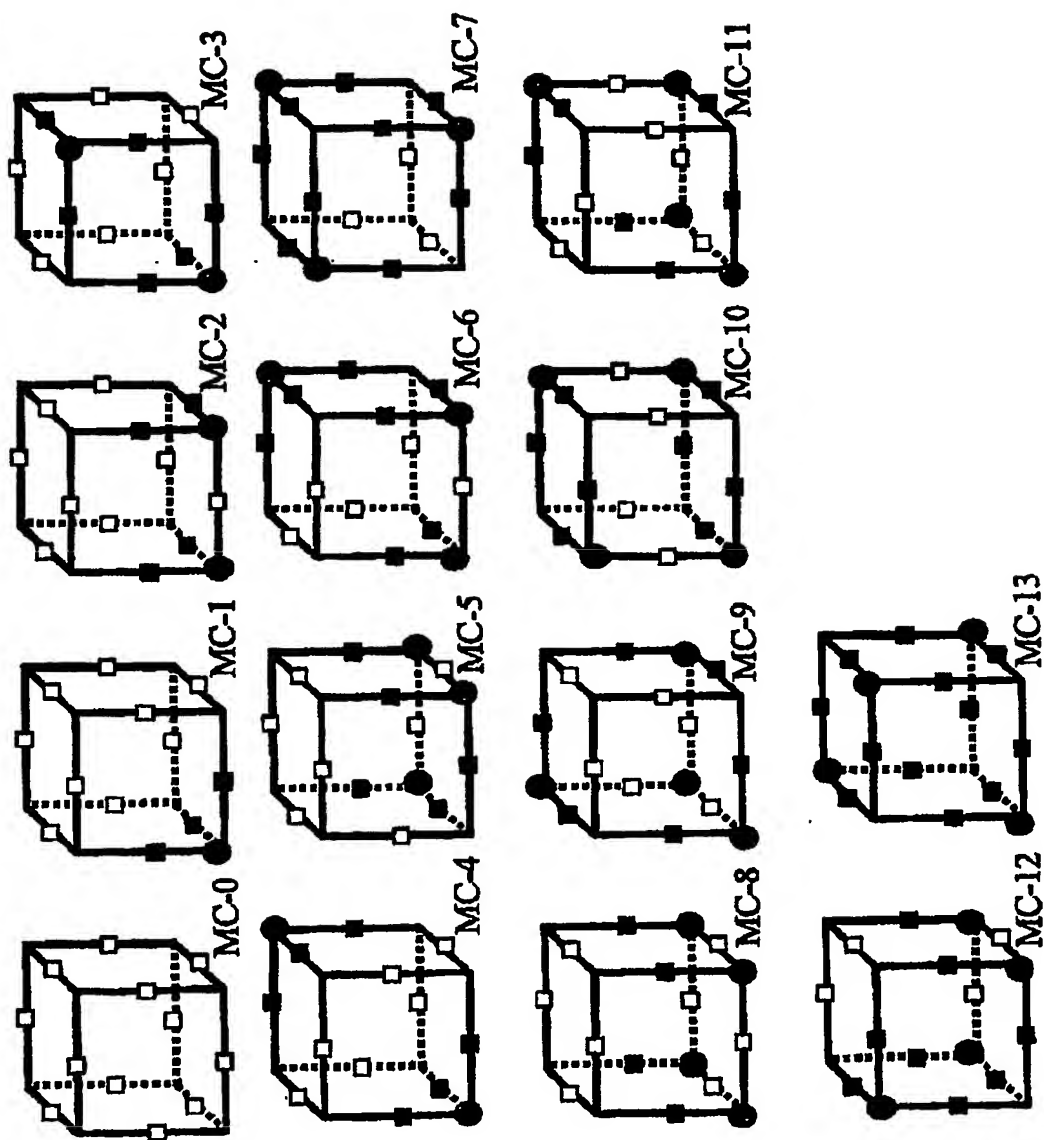


Fig.12: Reconstruction using partitions of the closed loop type.
B-rep data: (a) A cyclide surface, (b) A die from TSUBAMEX.
Reconstructions with partitions inside (c) 64*64*64 cells and (d) 512*512*512 cells

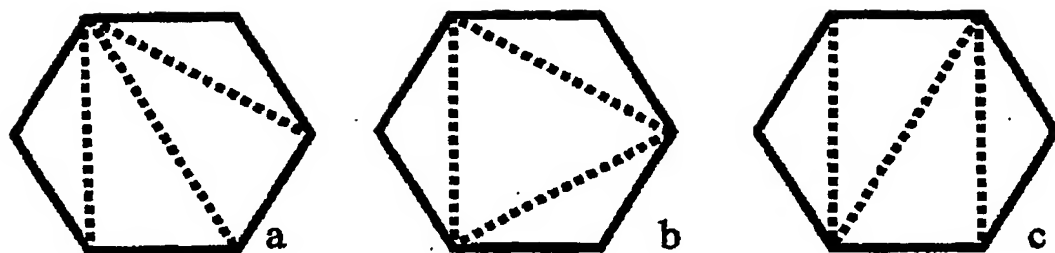
【図 16】



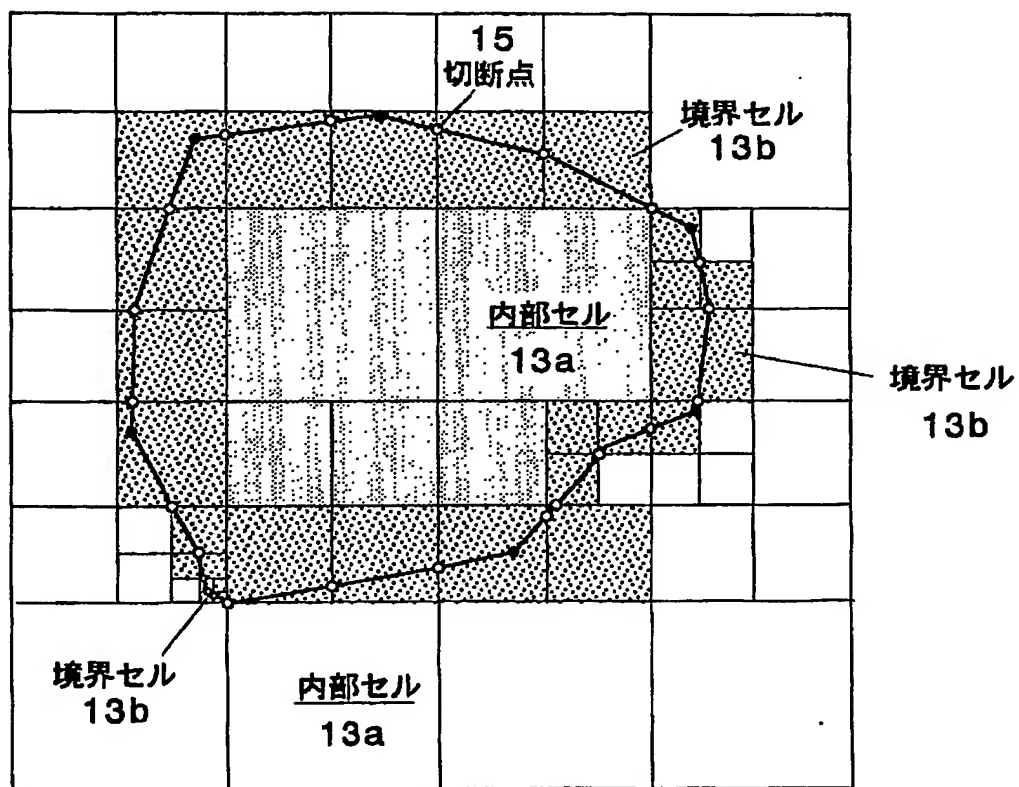
【図 17】



【図 18】



【図 19】



【書類名】 要約書

【要約】

【課題】 Marching Cubes (MC) によるすべてのセル稜切断点を内含しかつMCでは得られないすべてのセル稜切断点を漏れなく網羅することができ、これによりセル稜切断点を結ぶ境界からなるセル内形状データを漏れなく網羅することができる境界データのセル内形状データへの変換方法及び変換プログラムを提供する。

【解決手段】 対象物の境界データからなる外部データ12を境界が直交するセル13に分割する分割ステップ(A)と、境界データとセルの稜との交点をセル稜切断点とする切断点決定ステップ(B)と、求めたセル稜切断点を結ぶ境界をセル内形状データとする境界決定ステップ(C)と分割された各セルを対象物の内側に位置する内部セル13aと境界データを含む境界セル13bとに区分するセル区分ステップ(D)と、境界セルを構成するボクセルデータをセル内形状データより内側の内部ボクセルデータと外側の外部ボクセルデータに区分する境界ボクセルデータ区分ステップ(E)とを有する。

【選択図】 図1

出 願 人 履 歴 情 報

識別番号 [000006792]

1. 変更年月日	1990年 8月28日
[変更理由]	新規登録
住 所	埼玉県和光市広沢2番1号
氏 名	理化学研究所

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.